

AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A device for controlling a plurality of endpoints of a USB device, the device comprising:

a plurality of buffers allocated to the plurality of endpoints, respectively; and

an endpoint buffer controller for managing an exchange of packets between a host and the USB device, obtaining buffer-utilization information for each of the endpoints and adaptively adjusting at least one of:

a number of the buffers allocated to each of the endpoints based upon the buffer utilization, and

respective buffer capacities of the buffers allocated to each of the endpoints based upon the buffer utilization, respectively.

2. (Original) The device of claim 1, wherein each of the plurality of buffers has a plurality of units and a maximum size of $\text{unit_size} \times Z$, where Z is a positive integer representing the total number of units per buffer, respectively.

3. (Original) The device of claim 1, wherein the endpoint buffer controller includes:
a buffer status detecting section for determining whether an OUT packet from the host may be accommodated based on status information maintained for the FIFO buffers, and for generating a NAK when the OUT packet cannot be accommodated;

a timer for generating a NAK count reset signal for every T period;

a NAK counter for counting how many NAKs are generated for each of the endpoints within a period; and

a pointer control section for generating an interrupt signal when the respective NAK count

for one or more of the endpoints exceeds a threshold value.

4. (Original) The device of claim 3, further including a threshold control section for setting the threshold value.

5. (Original) The device of claim 4, wherein the endpoint buffer controller further includes a peripheral processor MCU, the MCU adaptively changing numbers of buffers allocated to the endpoints depending on the buffer-utilization information and storing the reallocated numbers in a register set.

6. (Original) The device of claim 3, further comprising:
a timer to determine the period T by doing one of taking the inverse of a frequency determined by counting SOF (Start Of Frame) signals from the host during an interval and adopting a frame period of a full-speed mode of a USB protocol.

7. (Original) The device of claim 3, wherein the pointer control section includes:
a NAK counter register set for storing the NAK counts for the endpoints;
a comparator for comparing the NAK counts stored in the NAK counter register with the threshold value, respectively;
a maximum packet size register set for defining a maximum packet size for each of endpoints; and
a buffer size register set for defining a size for each of the endpoints.

8. (Original) The device of claim 1, wherein the buffers are first-in, first-out (FIFO) buffers.

9. (Previously Presented) A method of controlling a plurality of endpoints of a USB device, the method comprising:

initializing buffers of each of the endpoints;

counting, when data are received at each of the endpoints, numbers of NAKs generated during a period T, respectively;

comparing the NAK counts with a threshold value; and

adaptively adjusting at least one of:

a number of the buffers allocated to each of the endpoints based on the NAK counts when one or more of the NAK counts exceeds the threshold value, and

respective buffer capacities of the buffers allocated to each of the endpoints based on the NAK counts when one or more of the NAK counts exceeds the threshold value.

10. (Original) The method of claim 9, wherein the period T is obtained by doing one of taking the inverse of a frequency determined by counting SOF (Start Of Frame) signals from the host during an interval and adopting a frame period of a full-speed mode of a USB protocol.

11. (Original) The method of claim 9, further comprising:

generating an interrupt when one or more NAK counts exceeds the threshold value; and

changing numbers of the buffers allocated to the endpoints based on the NAK counts, respectively, when the interrupt is generated.

12. (Original) The method of claim 9, further comprising:
operating the buffers of the endpoints of a first-in, first-out (FIFO) basis.
13. (Previously Presented) A configuration of a USB device comprising:
a serial interface engine (SIE) operable as an interface to a USB host;
a controller interface operable as an interface to a controller of the USB device; and
a buffer section to buffer at least one of information transferred from the SIE to the controller interface, the buffer section including a plurality of buffers corresponding to a plurality of endpoints, respectively,
at least one of:
a number of the buffers being allocated to each of the plurality of endpoints based upon buffer-utilization information, and
respective buffering capacities of the plurality of buffers allocated to each of the plurality of endpoints being adjusted based upon the buffer-utilization information.
14. (Original) The USB device configuration of claim 13, further comprising:
an endpoint buffer controller to adaptively allocate the respective buffering capacities of the plurality of buffers based upon buffer-utilization information.
15. (Original) The USB device configuration of claim 13, wherein:
each of the buffers is organized as a plurality of buffering units, respectively; and
each of the buffering units of a buffer includes a plurality of blocks of storage space, respectively.

16. (Original) The USB device configuration of claim 13, wherein each of the plurality of buffers is arranged to operate as a first-in, first-out (FIFO) buffer.

17. (New) A device for controlling a plurality of endpoints of a USB device, the device comprising:

a plurality of buffers allocated to the plurality of endpoints, respectively; and

an endpoint buffer controller for managing an exchange of packets between a host and the USB device, obtaining buffer-utilization information for each of the endpoints and adaptively adjusting:

a number of the buffers allocated to each of the endpoints based upon the buffer utilization.

18. (New) The device of claim 17, wherein the endpoint buffer controller adaptively adjusts respective buffer capacities of the buffers allocated to each of the endpoints based upon the buffer utilization, respectively

* * * * *

END OF CLAIM LISTING